LISTING OF PENDING CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A processor, comprising:

a module configured to collect status data from media access ~~control~~ devices connected to a bus, the status data indicating readiness of the media access ~~control~~ devices to participate in data transfers, the status data comprising data indicating whether a one of the media access control devices has received packet data;

one or more processing engines to schedule transfers of ~~data packets~~ packet data; and

a push engine to perform unsolicited transfers of the status data to the processing engines in response to the module collecting new status data.

2. (Currently Amended) The processor of claim 1, wherein the processing engines comprise:

one or more input transfer registers to receive the unsolicited transfers of status data for use to schedule the transfers of ~~data packets~~ packet data.

3. (Currently Amended) The processor of claim 2, wherein the processing engines use a portion of received new status data to schedule retrievals of ~~data packets~~ packet data from the devices.

4. (Currently Amended) The processor of claim 2, wherein the processing engines use a portion of the received status data to schedule transmissions of ~~data packets~~ packet data.

5. (Currently Amended) The processor of claim 4, wherein the processing engines use a portion of the received status data to determine whether schedule transmissions of ~~data packets~~ packet data have been completed.

6. (Original) The processor of claim 1, wherein the module is configured to poll the devices for the status data over a second bus.

7. (Currently Amended) The processor of claim 2, wherein a portion of the status data are flags indicative of whether associated devices have ~~data packets~~ packet data to transmit.

8. (Currently Amended) The processor of claim 2, wherein a portion of the status data includes flags indicative of whether associated devices have space to receive ~~data packets~~ packet data.

9. (Currently Amended) A method of transferring data packets over a bus, comprising:

collecting information on readiness of media access ~~control~~ devices connected to the bus to one of transmit and receive data packets; and

transferring a portion of the collected information to a processing engine configured to schedule data transfers, the transferring being unsolicited by the processing engine.

10. (Original) The method of claim 9, further comprising:

scheduling data transfers with a portion of the devices based on the transferred portion of the collected information.

11. (Original) The method of claim 10, wherein scheduling further includes:

determining whether the transferred information is at least partly new; and

wherein the scheduling is performed in response to the transferred information being at least partly new.

12. (Original) The method of claim 10, wherein determining includes comparing a value of a time stamp transferred with the information to a previous value of the time stamp.

13. (Original) The method of claim 10, wherein scheduling further comprises:

determining whether an earlier scheduled data transfer have been completed from the transferred information.

14. (Original) The method of claim 10, wherein collecting further comprises:

polling the devices for ready status data on the availability of ports thereon; and

receiving ready status data associated with individual ones of the devices in response to the polling.

15. (Previously Amended) The method of claim 12, wherein collecting further comprises:

writing the received ready status data to a status register; and

scheduling transfers of data packets over the bus in response to the transferred portion of the ready status data.

16. (Original) The method of claim 9, wherein the transferred portion of the information includes flags that indicate whether associated ports of the devices have one of space to receive data packets and data packets ready to transmit over the bus.

17. (Original) The method of claim 16, further comprising:

polling the ports of the devices over a second bus to determine values of the flags.

18. (Currently Amended) A router, comprising:

a bus; and

a parallel processor coupled to the bus and comprising:

a plurality of processing engines to process data transfers with a plurality of media access ~~control~~ devices connected to the bus; and

an interface connected to collect ready status data from the media access ~~control~~ devices and to automatically transfer ready status data to the processing engines in response to the status data being collected, the ready status data indicating readiness of the devices to participate in data transfers, the ready status data comprising data indicating whether a one of the media access ~~control~~ devices has received packet data.

19. (Original) The router of claim 18, wherein the ready status data indicates the readiness of individual ones of the devices to one of receive a data packet from and transmit a data packet to the parallel processor.

20. (Original) The router of claim 18, wherein the ready status data includes a time stamp indicative of a staleness of the ready status data.

21. (Original) The router of claim 18, wherein a portion of the ready status data includes information to enable the processing engines to identify which scheduled data transfers to the devices have been completed.

22. (Original) The router of claim 18, further comprising:

a ready bus capable of transferring ready status data from the devices to the interface.

23. (Original) The router of claim 19, wherein the ready status data indicates whether associated ports of the devices are ready to perform one of a transmission of a data packet to the bus and a receive of a data packet from the bus.

24. (Original) The router of claim 20, wherein each processing engine comprises at least one input transfer register; and

the interface is configured to write ready status data to one of the input transfer registers assigned to a scheduler thread.

25. (Original) The router of claim 24, wherein the interface is configured to protect one of the input transfer registers from being read by the processing engines during the transferring of ready status data thereto.

26. (Original) The router of claim 18, wherein the devices are capable of transmitting data packets between the bus and external networks.

27. (Original) The router of claim 18, wherein the interface transfers the collected status data without being solicited to transfer the data by the processing engines.

28. (Original) An article comprising a computer-readable medium which stores executable instructions for transferring data packets over a bus, the instructions causing a processor to:

collect information on readiness of media access ~~control~~ devices connected to the bus to one of transmit and receive data packets; and

transfer a portion of the collected information to a processing engine configured to schedule data transfers, the transferring being unsolicited by the processing engine.

29. (Original) The article of claim 28, the instructions further causing the processor to:

schedule data transfers with a portion of the devices based on the transferred portion of the collected information.

30. (Original) The article of claim 29, the instructions further causing the processor to:

determine whether the transferred information is at least partly new; and

wherein instructions causing the processor to schedule are performed in response to determining that the transferred information being at least partly new.

31. (Original) The processor of claim 1 in which the processing engines schedule the transfer of data packets independently of the module collecting status data from the devices.

32. (Original) The processor of claim 31 in which the processing engines schedule the transfer of data packets from a device to the bus independently of the readiness of other devices to receive the data, and schedule the transfer of data from the bus to a device independently of the readiness other devices to send the data.

33. (Currently Amended) A processor, comprising:

multiple multi-threaded programmable processing engines, individual ones of the programmable processing engines having at least one register; and

an interface ~~to at least one media access control device, the interface~~ operationally coupled to the multiple programmable processing engines, the interface comprising ~~logic to~~:

at least one register; and

logic to:

collect status data of at least one ~~of the at least one~~ media access ~~control devices~~ device via a bus, the status data indicating whether the at least one media access ~~control~~ device has received packet data; and

~~transferring~~ perform a transfer, unsolicited by the programmable processing engines, of at least a portion of the collected status data stored in the at least one register of the interface to at least one register of the multiple multi-threaded programmable processing engines.

34. (Currently Amended) The processor of claim 33, wherein the at least one media access ~~control~~ device comprises an Ethernet media access ~~control~~ device.

35. (Currently Amended) The processor of claim 33, wherein the interface comprises a push engine, the push engine to perform ~~an~~ the unsolicited transfer of the status data from the at least one register of the interface to the at least one register of the multiple multi-threaded programmable processing engines.

36 (Currently Amended) The processor of claim 33, wherein the interface further comprises logic to:

collect status data indicating ability of the at least one media access ~~control~~ device to receive data to transmit; and

transfer packet data to the at least one media access ~~control~~ device.

37 (Original) The processor of claim 33, further comprising at least one memory controller to a Synchronous Dynamic Random Access Memory (SDRAM).

38 (New) The processor of claim 33, wherein the interface further comprises a buffer to store packet data received by the at least one media access device.

39. (New) The processor of claim 33, wherein the at least one media access device comprises multiple media access devices.

40. (New) The processor of claim 33, wherein the status data of multiple media access devices is stored in a single one of the at least one register of the interface.